An OLIF-based open inflectional resource and yet another morphological system for German

Clematide, S

DOI: https://doi.org/10.1515/9783110211818.3.183

An OLIF-based open inflectional resource and yet another morphological system for German

Simon Clematide

Abstract. This paper describes the implementation of finite-state based, high precision morphological tools for the generation and analysis of open word classes based on the inflection classes for German of the Open Lexicon Interchange Format (OLIF). Productive compounding and derivations are treated by simple word formation rules. The latter is constrained by selective frequency checks over the web and corpora. Minimal lexicographic requirements (only stem and a numeric inflectional code) allow simple expandability and define a morphological abstraction layer which existing finite state morphological systems do not exhibit. Although a lot of lexical information is freely available for end users over the web, the same is not true for resources which will be used in NLP applications. Therefore, we initiate an open and shared morphological OLIF-based resource where we integrate material from sources which allows for such a term of use.

1 Introduction

The acquisition of morphological resources is commonly viewed as expensive in terms of "expert knowledge and labour" (Demberg 2007). Well, in fact it is expensive if it is done again and again by different academic researchers without sharing their resulting resources, and even more important, without a well-thought and well-agreed standard classification system which covers the needs of common text technology systems. For a highly inflected language as German, lemmatization and generation of inflected word forms is crucial for almost any text technological application. Simple and clear-cut interfaces for the coupling and extension of morphological and lexical resources are vital and should be based on standardized linguistic data categories. The EAGLES specification for German morpho-syntax (cf. EAGLES 1996) provides such a resource. For inflectional classes (in a very broad sense), the OLIF (Open Lexicon Interchange Format)[1] consortium has provided a list of "Recommended Values for OLIF Data Categories" for several languages including German (McCormick et al. 2004). In section 2, we described some more recent systems for German morphology. In section 3, we present our work in implementing a finite-state based morphological framework based on a minimal, but standard-oriented lexicographic interface.

1. See `http://www.olif.net`.

2 Other works

Perera and Witte (2005) have built a self-learning system called `DurmLemmatizer` that induces a German full-form lexicon for nouns by processing raw text corpora[2]. Their linguistic processing is embedded in the GATE framework (cf. Cunningham et al. 2002) and restricted to a standard part-of-speech tagger (TreeTagger), a base NP chunker (JAPE), and their own case and grammatical number tagger based on Hidden Markov Model. Lemmatization is done by stripping off native German inflection suffixes, therefore plural forms involving umlaut as in "Ärzte (pl); Arzt (sg)" (*doctor*) can't be treated correctly. In these and some other difficult cases, their algorithm inserts alternative possible lemma forms to gain recall (e.g. the possible lemma "*Öfen","*Öfe", "*Öf" (*oven*)). These alternative forms may be reduced, if a further analysis appears with only one of the previously possible lemma forms. An assessment of the quality of the lemmatization based on this resource is more difficult than it may seem. Firstly, the evaluation results in their paper is based on a rather small lexicon with about 13'000 entries whereas the currently distributed resource contains about 84'000. Secondly, their own evaluation numbers need careful interpretation. They are gained against a subset of 88% of all noun occurrences where the TreeTagger was also able to produce a lemma. About 75% of the noun occurrences thereof are lemmatized by their system with a precision of around 95%. However, it's unclear how they treat cases where the lexicon contains alternative lemmas – the current distribution of their lexicon has about 14'000 ambiguous lemmatizations.

The SOAP services from `http://wortschatz.uni-leipzig.de` allow the request for the generation of other word forms from a given one. This service is described as "For a given word form returns all other word forms of the same lemma". The word form "geben" (*to give*) produces the output "gibt gab geben gegeben gebe gaben gäbe gibt's Gibt gab's" which makes obvious that only forms which are covered in the corpus are returned. The word form "lieben" (verb *to love* or adjective *dear*) seems to return adjective forms only: "lieber liebsten lieben liebe lieb liebste liebstes liebes liebster liebstem". Although a verb and an adjective reading is returned by their base form service.

Geyken and Hanneforth (2006) present their German morphological analyzer based purely on finite state methods with weighted transitions[3]. The architecture of this system basically allows free combinations of the items from their stem (80'000 entries) and affix lexicon. About 1'000 morphotactic constraints (word grammar) restrict the possible combinations according to the language specific rules and limit morphological overanalyses. However, there are still lots of unwanted and irrelevant though possible morphological segmentations which one would like to get rid off.

2. `http://www.ipd.uka.de/~durm/tm/lemma/`
3. An online demo is available from `http://www.tagh.de`.

With the use of penalty weights associated with morphological boundaries and rare morphemes, an optimality ranking between competing analyses emerges from the analyses itself. Volk (1999) showed in the context of GERTWOL (cf. Koskeniemmi and Haapalainen 1996) that the heuristic "prefer simple analyses" is very effective in determining the intended lemma. Without weighted automata, one has to do this in a separate postprocessing filter.[4] Still, the weighted automata do not suppress unwanted analyses. The TAGH stem lexicon consists of complex entries because every stem alternation gives raise to a separate entry: E.G. the German verb "werfen" (*to throw*) needs the following lemma-stem pairs "werf:warf", "werf:werf", "werf:wirf", "werf:worf", "werf:würf" with their corresponding morphological features which determine the distribution of the stems in the inflectional paradigm. But there is also a lot of redundancy in this entries for the information which belongs to the lemma itself. The following two entries for past and past participle illustrate this point.

```
(werf:warf) [VIRREG VType=main PrefVerb=no Latinate=no StDef=yes St23SgInd=no
             StPret=yes StSubjI=no StSubjII=no StPartII=no StImpSg=no
             St23SgIndVowelChange=yes]
(werf:worf) [VIRREG VType=main PrefVerb=no Latinate=no StDef=yes St23SgInd=no
             StPret=no StSubjI=no StSubjII=no StPartII=yes StImpSg=no
             St23SgIndVowelChange=yes]
```

The TAGH system is optimized towards coverage.[5] For the 100 million word corpus "DWDS-Kerncorpus"[6] the authors give a coverage of 98.2%. Although no published quantitative evaluation on the correctness of the analyses is available, its effective use in two large scale and public lemmatization applications grants high quality.

Schmid et al. (2004) present a morphological analyser that recognizes derivation and composition. Stems may therefore be basic, derived or compounds. Affixes have the origin classes native, foreign, classical. They select their stems by word class features. An illustrating extract from the SMOR lexicon included in the SFST software distribution is shown below:

```
<Base_Stems>haus<PREF>:<><ge>ha:i<>:elt<V><base><nativ><VVPastStr>
<Base_Stems>haus<PREF>:<><ge>ha:ält<V><base><nativ><VVPres2t>
<Base_Stems>haus<PREF>:<><ge>halt<V><base><nativ><VVPP-en>
<Base_Stems>haus<PREF>:<><ge>halt<V><base><nativ><VVPres1>
<Base_Stems>g:bu:et:<><ADJ><base><nativ><AdjSup>
```

4. Such a post-processing filter has an extreme low memory and processing cycle footprint if it's done using a standard UNIX flex tool as our own reimplementation of the original PERL code shows.
5. However, on the demo web site they mention that rare word form (a threshold of 10 over a corpus of 500 million tokens) are omitted for efficiency reasons.
6. `http://www.dwds.de`

```
<Base_Stems>g:bu:et:s<>:s<ADJ><base><nativ><AdjComp>
<Base_Stems>Roß:s<>:s<>:e<NN><base><nativ><NNeut/Pl>
```

The entries include structural (<PREF> "prefix"), morphotactic (<nativ>) and inflectional (<VVPres2t>) information. As in the case of TAGH, each stem alternation (e.g. a:i) is encoded by a separate lexicon entry. This is also true for suppletive gradation as "gut" (good), "besser" (better).

3 Architecture of mOLIFde

Other than the discussed SMOR or TAGH systems, our morphological system has minimal requirements for the lexicographic interface: An atomic stem[7] and an OLIF inflection code: E.G.

```
haus|halt 387
obig 531
Reichtum 111
```

For our internal lexical grammar, we strictly follow the EAGLES specification for German morpho-syntax (EAGLES 1996) which grants us compliance with STTS (Schiller et al. 1999) and documentation. We use the morpho-syntactic features and values verbatim (e.g. "&pos" "=noun"[8]) and serialize them top down according to the hierarchy presented in the standard. The raw EAGLES format and its corresponding shorter STTS representation look like

```
Reichtümern  &pos=noun&type=com&declin=no&numb=pl&case=dat&gend=masc&infl=--
Reichtümern  NN:Masc.Dat.Pl.*
```

3.1 The struggle with OLIF inflection categories

The recommended OLIF data categories for inflection codes contain more than 700 quite fine-graded word classes. For the open inflectional word classes, we find the following numbers: verbs (388), nouns (216), adjectives (34). These classes are more or less directly taken from the LOGOS machine translation system (cf. Scott 2004). To our knowledge, other lexical standardization initiatives (e.g. ISLE/MILE (Ide et al. 2003)) have not produced data category sets comparable to this list. Fig. 1

7. The only exception is a boundary marker after separable verb prefixes that marks also the place for the insertion of "ge" in past participles.
8. For a concise documentation on the syntax of the Xerox regular expression calculus see http://www.xrce.xerox.com/competencies/content-analysis/fsCompiler/fssyntax.html.

displays an extract of the noun inflection codes. Roughly said, they define a morphological abstraction layer which also covers some lexical and distributional informations needed for common text technological applications. Although the number of classes may be seen as high, coverage is not perfect.[9]

OLIF systematically shows separate classes for root verbs ("handeln" *to trade*), verbs with inseparable prefix ("behandeln" *to treat*), verbs with separable prefix ("herunterhandeln" *to beat down*), and verbs with a separable and an inseparable prefix ("wiederbehandeln" *to treat again*). The latter are quite uncommon as finite forms, however, adjectival use of past participles built out of them or nominalizations are more frequent. The German dictionary WAHRIG (Wahrig and Wahrig-Burfeind 2006) contains a list of 188 inflection paradigms for strong verbs, which would lead to an upper limit of 752 verb classes.

The high number of noun classes is mostly due to foreign words with foreign or alternate inflection paradigms ("Klima" *climate*, with 3 plural forms in nominative plural as "Klimata", "Klimate","Klimas") and the fact that every OLIF class has its determined gender even with identical inflection (e.g. "Vater" (*father*) masculine 51, "Kloster" (*convent*) neuter 141). There is also suppletive plural formation (e.g. the plural "Streuzuckersorten" for the uncountable German "Streuzucker" (*castor sugar*)) which may be practical for machine translation systems, but may seem idiosyncratic otherwise. Additional classes evolve from nouns with singular or plural forms only. Nouns with alternate paradigms get their own OLIF class which may lead to many additional classes when done consequently. Another more lexicographic question arises with nouns with alternate gender (often attributed to regional preferences, e.g. the masculine form "Gehalt" used in Austria in the sense of salary in contrast to the standard German neuter gender). And last but no least, spelling reforms of German have produced additional classes.

The linguistic characterisations of the different OLIF inflection classes are often sparse, as can be seen in Fig. 1. The use of arbitrary numbers as class identifiers may seem odd at first. The use of prototype lemmata in the style of "inflects like" should give a intuitive access to the classes. Still, an explicit explanation about the intended sense of a class would have made our work a lot easier. The example lemma itself may also be a source of confusion. For example, OLIF has an inflection class 105 `-s/-"e` exemplified by the lemma "Sonnenbrand" (*sunburn*) which therefore disallows "*Sonnenbrandes"[10]. Neither the Canoo language tools[11] nor WAHRIG support this limitation, and an exact Google search gives about 2'000 hits for "Son-

9. Unfortunately, the integer IDs for the classes are not even unique across different part-of-speech. On the other hand, there are quite a few classes which are redundant, i.e. they cover the same phenomena.
10. There exists a noun class 55 "Wunsch" (wish) `-es/-"e` that seems to enforce schwa in genitive singular.
11. `http://www.canoo.net`

POS	Gender	Example	Inflects Like	Code
noun	feminine	Mutter	-/-" like Mutter/Mütter	53
noun	feminine	Hand	-/-"e like Hand/Hände	57
noun	feminine	Frau	-/-en like Frau/Frauen	64
adjective		arm	With umlaut and st in superlative like arm, ärmer, ärmst	96
verb		herausschinden	Irregular with separable prefix, like herausschinden - herausschund - herausgeschunden	645

Figure 1. Information contained in the OLIF inflection classes for German

nenbrands", but 8'000 for "Sonnenbrandes". There exist quite a few classes with overlapping or identical extension. The decision whether there is real redundancy has to be done painstakingly. In short, OLIF inflection codes were not as perfect as initially imagined. Along our development, we detected various problems and omissions which the OLIF consortium used to correct things according to our feedback.

3.2 Our word-and-paradigm finite state morphology

Our system is implemented using the Xerox finite state tool `xfst` (cf. Beesley and Karttunen 2003). The benefits of transducers for morphology systems are common place now: Bidirectionality (generation and analysis), non-determinism (regular relations encode many-to-many mappings, i.e. a word form allows more than one analysis and the same morphological features may produce more than one word form), efficiency in processing time and memory.

One special feature of our system is the ability to generate word forms in a class based fashion. Our demo web service[12] generates any desired inflectional paradigm for a given lemma by specifying the corresponding OLIF inflection class. Though monolithic morphologic systems as SMOR or TAGH can generate, they are limited to their lexical content which can't be extended simply by a pair of stem and inflection class.

Finite state morphology engineering is either based on a two-level rule component as GERTWOL (Koskeniemmi and Haapalainen 1996), or on composition of replacements and restrictions since the invention of the replacement operator (Karttunen 1995). We decided to use the latter serial approach because our lemma lexicon does not contain stem alternation, and therefore a lot has to be done by rules to ensure the correct word forms.

12. See `http://www.cl.uzh.ch/kitt/molif` for morphological generation and analyses.

```
|A ,B ,C ,D ,E ,F ,G ,H ,I ,J ,K ,L, OLIFC  example     |A B  B  D  E  F  G OLIFC, STEMRULE, example
 A0,B0,C0,D0,E0,F0,G0,H0,I0,J0,K0,L0, 90 | ‘‘klein‘‘    A4,BX,CX,DX,EX,FX,G0,  1|, SEIN, ‘‘sein ‘‘
 A0,B0,C0,D0,E0,F2,G0,H0,I0,J0,K0,L0, 96 | ‘‘sicher‘‘   A3,B1,C9,D0,E1,F2,G2, 83|, WACHSEN, ‘‘auf|wachsen‘‘
 A0,B0,C0,D0,E0,F0,G1,H0,I0,J0,K0,L0,132 | ‘‘arm‘‘      A0,B0,C0,D0,E0,F0,G0,  4|, MACHEN, ‘‘machen‘‘
 A0,B0,C0,D0,E1,F0,G0,H0,I0,J0,K0,L0,135 | ‘‘dunkel‘‘   A3,B0,C1,D4,E0,F1,G0,285|, SCHNEIDEN, ‘‘schneiden‘‘
 A0,B0,C0,D0,E0,F0,G2,H0,I0,J0,K0,L0,422 | ‘‘schmal‘‘   A3,B0,C1,D4,E0,F1,G1,286|, SCHNEIDEN, ‘‘beschneiden‘‘
 A0,B0,C0,D3,E0,F0,G0,H0,I0,J0,K0,L0,446 | ‘‘gut‘‘      A3,B0,C1,D4,E0,F1,G2, 39|, SCHNEIDEN, ‘‘an|schneiden‘‘
 A1,B0,C1,D0,E0,F0,G0,H0,I0,J0,K0,L0,522 | ‘‘rosa‘‘     A3,B0,C1,D4,E0,F1,G3, 49|, SCHNEIDEN, ‘‘mit|beschneiden‘‘
 A0,B1,C1,D0,E0,F0,G0,H0,I0,J0,K0,L0,531 | ‘‘obig‘‘
```

Figure 2. Extract of the matrix of linguistic features for adjectives and verbs

The huge number of inflection classes which had to be managed required a systematic specification approach with as much as possible automated reuse thereof. In the first place, every OLIF inflection class had to be reconstructed as a matrix of linguistic features.

Figure 2 shows some sample feature vectors. For adjectives, we have e.g. A0 = flectional, A1 = non-flectional; B0 = attributive and/or predicative use, B1 = attributive use only; C0 = unlimited gradation, C1=positive only; D3=irregular gradation stem; F2=optional elision of e in comparative forms; G0=no umlaut, G1=umlaut, G2=optional umlaut.

For verbs, we have e.g.: A=main verb class: A0=regular A4=special inflection A3=strong verb; B=special present forms: B0=no umlaut B1=umlaut; C=ablaut in past and past participle: C0=no change C1=ei-i-i C9=a-u-a; D: additional stem changes (consonant): D0=no change D4=d-tt; E=umlaut in past subjunctive: E1=normal umlaut; F=final sound classes: F1= dental (-d,-t) F2=sibilant (-s,-z) without -sch; G=verb prefix: G0=no prefix G1=inseparable prefix G2=separable prefix G3=both prefixes.

The inflection component for each adjective class has an architecture as depicted in Fig. 3. Similar architectures are used for verbs and nouns. In order to keep the manual writing of class-specific replacement rules consistent and short, two mappings are automatically built by processing the feature matrix.

- Feature macros (e.g. `AdjectiveMacroG2`) contain the union of every OLIF class tag exhibiting the corresponding feature. The restriction concerning attributive use can be written as:

```
define AdjectiveUseRestr [
    "&use"  => .#. [$. AdjectiveMacroB1] _ "=attr"
             ,  .#. [$. AdjectiveMacroB2] _ "=nattr" ];
```

- Class rules (e.g. `AdjectiveRule135`) contain the composition of general restrictions together with all the class specific feature rules (`AdjectiveRuleE1`) which have to be coded manually. The rule for feature E1 (deletion of “e” in attributive positive and comparative forms in lemmata as “dunkel” (dark)) looks

like[13]:

```
define AdjectiveRuleE1 [
     [ $ [{el} "<DEGR/>"] ]  # precondition: ensure lemma is ending on -el
 .o. [ e -> 0 || _ l "<DEGR/>" ["<COMP/>"|"<POS/>"][$. ["&use" "=attr"]]]];
```

The precondition ensuring "-el" is only necessary for keeping generation of paradigms specific and discriminating, because it excludes any stem with feature E1 not ending on "-el" from producing word forms. This is essential if we try to induce the OLIF inflection class from full form lexica.

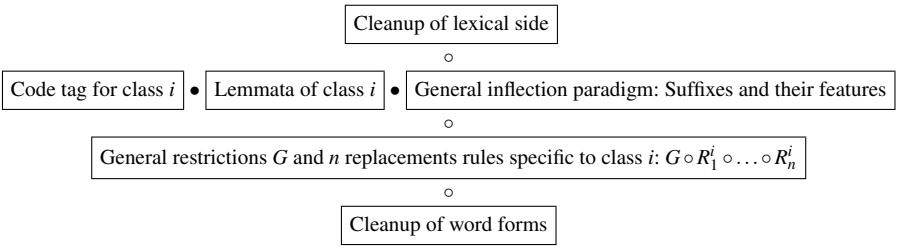

Figure 3. Main architecture of the mOLIFde inflection component for a single inflection class: ∘ means composition, • concatenation.

For class based generation, we need to keep the composed replacement rules separated from the lexicon. The composition of replacement rules (which are typically cyclic and reentrant) can quickly lead to huge transducers and long compilation times. A careful explicit definition of the lexical language and its composition to the rules has been proven critical to reach our goals. The compiler needs some hints where morphological values may appear and where they won't. An extreme example is the purely rule-based treatment for the stem "sein" (be) where we additionally to stem changes specify the real inflection paradigm.

```
...
.o. [{sei} "<VINFL/>" {en} -> {wär} "<VINFL/>" e        || _ ?* "&fin" "=fin"
     "&vf-m" "=konj" "&numb" "=sg" "&pers" ["=1"|"=3"] "&tense" "=past"]
.o. [{sei} "<VINFL/>" {en} -> {wär} "<VINFL/>" (e) {st} || _ ?* "&fin" "=fin"
    "&vf-m" "=konj" "&numb" "=sg" "&pers" "=2"  "&tense" "=past"]
.o. [{sei} "<VINFL/>" {en} -> {wär} "<VINFL/>" {en}     || _ ?* "&fin" "=fin"
    "&vf-m" "=konj" "&numb" "=pl" "&pers" ["=1"|"=3"] "&tense" "=past"]
.o. [{sei} "<VINFL/>" {en} -> {wär} "<VINFL/>" (e) t    || _ ?* "&fin" "=fin"
    "&vf-m" "=konj" "&numb" "=pl" "&pers" "=2" "&tense" "=past"]
...
```

13. The gradation suffixes "er" and "st" are represented internally by abstract morphemes "<COMP/>" and "<POS/>" and realized in the cleanup step of the word form. This keeps the size of the composed transducers reasonable.

Without the lexical language the resulting transducer which generates all inflectional and non-inflectional forms, `xfst` gives the following properties: 110.1 Mb. 214706 states, 8886612 arcs, Circular. Composing the lexical language drastically reduces compilation time and size: 151.5 Kb. 812 states, 11507 arcs, Circular. Although this may still seem big, further composition with the lexicon entry "sein" results in a normal lexical transducer: 8.7 Kb. 280 states, 307 arcs, 34 paths.

3.3 Derivation, conversion, and compounding

Our lexicon doesn't provide origin information as SMOR. In contrast to compounding, derivation is a bounded process. Therefore, we can easily produce all derived lemmas[14] and validate them afterwards by frequency checks over web-based search engines and corpora[15]. Applying a threshold to the frequency counts gives us quite reliable results, although no systematic evaluation has yet been done. In the current state, we derive all verb forms with separable prefixes from a list of around 100 prefixes. For the frequency checks of this verbs, the past participle is a good choice. Productive and regular derivations which we would like to treat properly appear often in iterated suffixation (adjectives ending on "-ig" derive nouns on "-igkeit"). The corresponding OLIF inflection classes of the source stem and the derived stem can be predicted with high precision.

Productive noun compounding is done as in SMOR with inflected forms (nominative singular and plural, genitive singular) for the first element using the inflection suffix as the linking morpheme. This has to be enriched by feminine noun classes with linking elements "-s-" that are not part of their inflectional paradigm, as well as some nouns as "Schule" (*school*) where final "e" is deleted as in "Schulhaus" (*school building*).

The problem of overanalyses introduced by compounding is also present in our system. Within the finite state calculus we implemented optionally a method called "lexicon prioritizing" to effectively remove overanalyses in the lexical transducer which are already covered by the lexicon. First, we determine a transducer that has all word forms of analyses from simple lexicon entries which can be reanalyzed by compounds on one side, and on the other side the corresponding compound analyses we want to suppress. Second, we use the side with the compound analyses to remove them from the lexical side of the original transducer. The calculation for this operations takes some minutes for the current lexicon size (see Fig. 4) and it's the most expensive compilation step regarding memory consumption and processing time.

14. Of course, conversion has also to be done. We have implemented a fix point computation that stops when conversion and derivation do not produce further new forms.
15. The SOAP services from `http://wortschatz.uni-leipzig.de` are very useful for this.

3.4 An open OLIF-based German lexicon

The lack of open and shared high-quality morphological resources adapted for the use in text technological applications is a dissatisfying situation for a language as German. Although, there is currently an interest in the automatic learning of morphological segmentation Demberg (2007)[16], the results in the `DurmLemmatizer` lexicon show the difficulties of purely data-oriented boot-strapping approaches.

When we decided to adhere to the OLIF inflection classes, we had the aim to find preclassified entries which could be easily integrated. One hope was the lexicon of the OpenLogos[17] translation system which contains a huge relational database and which was the original source of the OLIF inflection classes. Unfortunately, we had some problems to access it and to take it apart. Currently we are in the process of integrating and mass validating its 165'000 lemmas into our resources we converted in the meantime. The number of lemmas is high, because conversion results as nominalized infinitives and deverbal adjectives are separately listed.

In the first time, we used the full form lexicon which can be exported from the public, but closed source Windows-based system Morphy (Lezius 2000) to induce the inflection classes. Our morphology produced the possible paradigms for each stem, then we compared the results with Morphy's paradigm, and tried to identify a single class. In the course of this work, we found several omissions and errors on our side as well as some peculiarities how Morphy treats the rare past subjunctive forms of strong verbs. For about 21'000 noun lemmas, 5'500 adjective lemmas, 4'000 verbs lemmas (without separable verb prefixes) a single class was identified. One interesting point of this resource in terms of analyses coverage is the tendency of Morphy to postulate a lot of singulare tanta nouns and non-gradable adjectives – although in many cases, it's morphologically sound to produce plural or comparative forms. The restrictions stem from semanto-lexicographic determinations of the words which normally takes place when word forms are coupled with specific meanings. The same kind of frequency checks we use for the validation of derived word forms, can be used to check and quantify the tendency for restricted use of such words.

Third, we used open bilingual resources[18] and extracted adjectives and nouns with frequent and regular suffixes. Classification validation can be done more quickly this way.

16. `http://www.cis.hut.fi/morphochallenge2008`
17. `http://logos-os.dfki.de`
18. `http://www.dict.cc`

Category	Lexicon	Conversion	Derived	All
Verb	4745	0	17393	22138
Noun	20474	21987	15526	57987
Adjective	12173	43865	2997	59035
All:	37392	65852	35916	139160

Figure 4. The current distribution of lexical entries with derivative forms filtered by a threshold of 5 occurrences.

4 Conclusion

We think that a shared, simply extendable, and standard-based morphological resource for German fills a gap for text technology and lexicography. High precision lemmatization and generation of word forms should be standard techniques, self-learning systems may help to extend or optimize further. Huge and well supported corpora with application interfaces are an invaluable service therefore. The use of closed-source software for our morphological tools may seem inconsistent. However, our approach needed powerful and developer-friendly finite state tools already two years ago when the development started. For the finite part of the lexicon we have created an textual export into the open-source SFST tools. A current project will use our morphology in a web-service for generation of inflected forms for the automatic recognition of glossary entries in the OLAT[19] learning management system. [20]

References

Beesley, Kenneth R. and Lauri Karttunen (2003). *Finite-State Morphology: Xerox Tools and Techniques*. CSLI Publications.

Cunningham, Hamish, Diana Maynard, Kalina Bontcheva, and Valentin Tablan (2002). A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 168–175, University of Pennsylvania, URL http://www.aclweb.org/anthology/P02-1022.pdf.

Demberg, Vera (2007). A Language-Independent Unsupervised Model for Morphological Segmentation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 920–927, Prague, Czech Republic: Association for Computational Linguistics, URL http://www.aclweb.org/anthology/P/P07/P07-1116.

EAGLES (1996). ELM-DE: EAGLES Specifications for German morphosyntax: Lexicon Specification and Classification Guidelines. electronic, URL http://www.ilc.cnr.it/EAGLES96/pub/eagles/lexicons/elm_de.ps.gz.

19. http://www.olat.org

20. Thanks to Thomas Kappeler and Luzius Thöny for implementing the verb and noun part of the system.

Geyken, Alexander and Thomas Hanneforth (2006). *Finite-State Methods and Natural Language Processing, 5th International Workshop, FSMNLP 2005, Helsinki, Finland, September 1-2, 2005. Revised Papers*, chapter TAGH: A Complete Morphology for German Based on Weighted Finite State Automata, 55–66. Springer, URL http://dx.doi.org/10.1007/11780885_7.

Ide, Nancy, Alessandro Lenci, and Nicoletta Calzolari (2003). RDF Instantiation of ISLE/MILE Lexical Entries. In *Proceedings of the ACL 2003 workshop on Linguistic annotation*, 30–37, Morristown, NJ, USA: Association for Computational Linguistics, doi:http://dx.doi.org/10.3115/1119296.1119301.

Karttunen, Lauri (1995). The Replace Operator. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, 16–23, Cambridge, Mass, URL http://www.aclweb.org/anthology/P95-1003.pdf.

Koskeniemmi, Kimmo and Mariikka Haapalainen (1996). GERTWOL – Lingsoft Oy. In Roland Hausser (ed.), *Linguistische Verifikation : Dokumentation zur Ersten Morpholympics 1994*, number Band 34 in Sprache und Information, 121–140, Tübingen: Niemeyer.

Lezius, Wolfgang (2000). Morphy - German Morphology, Part-of-Speech Tagging and Applications. In Ulrich Heid, Stefan Evert, Egbert Lehmann, and Christian Rohrer (eds.), *Proceedings of the 9th EURALEX International Congress*, 619–623, Stuttgart.

McCormick, Susan M, Christian Lieske, and Alexander Culum (2004). OLIF v.2: A Flexible Language Data Standard. URL http://www.olif.net/documents/OLIF_Term_Journal.pdf.

Perera, Praharshana and Rene Witte (2005). A Self-Learning Context-Aware Lemmatizer for German. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*, 636–643.

Schiller, Anne, Simone Teufel, and Christine Stöckert (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). URL http://www.ims.uni-stuttgart.de/projekte/corplex/TagSets/stts-1999.pdf.

Schmid, Helmut, Arne Fitschen, and Ulrich Heid (2004). SMOR: A German Computational Morphology Covering Derivation, Composition, and Inflection. In *Proceedings of the IVth International Conference on Language Resources and Evaluation (LREC 2004)*, 1263–1266.

Scott, Bernard (Bud) (2004). The Logos Model: An Historical Perspective. *Machine Translation* 18:1–72, URL http://dx.doi.org/10.1023/B:COAT.0000021745.20402.59.

Volk, Martin (1999). Choosing the Right Lemma when Analysing German Nouns. In *Multilinguale Corpora: Codierung, Strukturierung, Analyse. 11. Jahrestagung der GLDV*, 304–310, Frankfurt.

Wahrig, Gerhard and Renate Wahrig-Burfeind (eds.) (2006). *Wahrig Deutsches Wörterbuch: mit einem Lexikon der Sprachlehre*. Gütersloh: Wissen Media Verlag, 8. edition.